Google

Web Images Groups News Froogle Local **more »**

event management and real time [Search] Advanced Search Preferences

The "AND" operator is unnecessary -- we include all search terms by default. [details]

Web Results **1** - **10** of about **143,000,000** for event management and real time. (**0.38** seconds)

Lucent - VitalSuite® Network Performance **Management** Software **Real ...**
VitalSuite® Network Performance **Management** Software **Real-time Event** Module for ... VitalEvent® **Real-time Event** Analysis software is a flexible, scalable, ...
www.lucent.com/products/solution/ 0,,CTID+2020-STID+10439-SOID+1260-LOCL+1,00.html - 32k - Cached - Similar pages

[PDF] **Real-Time** and Historical **Event Management**
File Format: PDF/Adobe Acrobat - View as HTML
Real-Time and. Historical **Event**. **Management**. Dell OpenManage™ ... Network Manager's **Event** Monitor displays a **real time** view of **events** as they are ...
www.dell.com/downloads/global/products/pwcnt/en/pwcnt_kyle_**events**.pdf - Similar pages

RFID Resource Center: **Real-time** Data Access and **Event** Processing ...
More RFID Data **Management** and **Event** Stream Processing Links. RFID Articles and Media Coverage ... The **Real Time** Data **Management** Imperative ...
www.progress.com/**realtime**/rfid_resources/index.ssp - 41k - Cached - Similar pages

Progress **Real Time** Division delivers **real-time** data **management** to ...
... of **real-time** data **management** products that enable database **management**, ... We deliver technologies that monitor and analyze **real-time event** stream data ...
www.progress.com/**realtime**/index.ssp - 26k - Sep 29, 2005 - Cached - Similar pages
[More results from www.progress.com]

IBM eServer Systems **Management** - IBM **Real Time** Diagnostics
Real Time Diagnostics is systems **management** software from IBM that runs on your IBM eServer ... System **Event** Log. •, Restart I/O Module Extended Diagnostics ...
www.ibm.com/servers/eserver/xseries/ systems_**management**/sys_migration/rtd.html - 24k - Cached - Similar pages

Online **event** registration for easy **event management**
... built in **event** registration, questionnaires and payment processing; **real-time** ... **Manage**. One of the most **time** consuming necessities of any **event** is ...
www.acteva.com/index.cfm - 20k - Sep 29, 2005 - Cached - Similar pages

Optimize Magazine > Business **Management** > **Real-Time** Information ...
Real-time event detection requires a rigorous process for identifying these. Discussions of **real-time** information tend to confuse **event** detection with the ...
www.optimizemag.com/article/ showArticle.jhtml?articleId=25600241 - 33k - Cached - Similar pages

InternetNews **Realtime** News for IT Managers
Create and **manage** accurate, **real-time** sales forecasts. ... Service Impact and **Event Management** Route to Value–A Path to Effective Business Service ...
www.internetnews.com/ - 70k - Sep 29, 2005 - Cached - Similar pages

eyefortransport "Using telematics and **real time** data **management ...**

... supply chain **event management** technology and dramatically increase your profits ...
Book now for the eyefortransport "Using telematics & **real time** data ...
www.eyefortransport.com/**realtimedata**/index.shtml - 21k - Cached - Similar pages

BMC Software: Service Impact **Management**
Real-time Service Impact **Management** Master Cell Business Enterprise Remedy Level ...
Service Impact **Management** (SIM) solutions extend advanced **event** ...
www.bmc.com/products/products_ services_detail/0,,0_0_0_703,00.html - 32k - Sep 29, 2005 - Cached - Similar pages

Result Page: 1 2 3 4 5 6 7 8 9 10 **Next**

event management and real time | Search

Search within results | Language Tools | Search Tips | Dissatisfied? Help us improve

Google Home - Advertising Programs - Business Solutions - Hurricane Resources - About Google

Operations Management

Unicenter CA-SYSVIEW Realtime Performance Management Event Capture Option

Monitoring and Managing Mainframe System Resource Performance

features at a glance

- Displays past performance information online
- Manipulate past performance displays for determining the root cause
- Captures performance information at the time of a performance problem
- Displays SMF records online
- Suppress or log defined SMF record types

Unicenter CA-SYSVIEW is a comprehensive, integrated, real time performance monitoring and management solution that provides powerful viewing and command facilities for the mainframe, CICS, IMS, UNIX System Services, and WebSphere MQ for the mainframe. It also provides valuable performance insight and status information of JES2, Parallel Sysplex and CA-DATACOM. The Unicenter CA-SYSVIEW Event Capture Option further enables users to troubleshoot past performance problems online. More...

product news

CA Addresses Cus
Evolving Mainfran
Challenges with G
Availability of Uni
Solutions

customer succe:

Southwest Airline
Standardizes on C
Mainframe Soluti
Reduce Costs and

more resources

> Sign up for E-N
> Email this page

information center

> FAQs

featured topic

Events Calendar - From Webcasts to Seminars to Trade Shows
CA has many informative and cutting edge events all year long

related solutions

Database Performance Management
The Solution for Optimal Database Performance

complementary products

Unicenter CA-OPS/MVS Event Management and Automation
Higher Levels of Automation = Higher Levels of Availability

Unicenter Performance Management Predictor for z/OS and OS/390
eBusiness Infrastructure Management

How valuable is this information? Not at all ○ ○ ○ ○ ○ Extremely **Submit**

Contact Legal Notice Privacy Policy Site Map

Computer

Web Images Groups News Froogle Local more »

event management and real time Search Advanced Search Preferences

The "AND" operator is unnecessary -- we include all search terms by default. [details]

Web Results **11** - **20** of about **143,000,000** for **event management and real time**. (**0.09** seconds)

TechRepublic - **Real** World, **Real Time**, **Real** IT
Upcoming live **event**. Webcast: IT Change **Management**: Complexity, Challenges and Technologies ... TechRepublic - **Real** World. **Real Time**. **Real** IT. ...
techrepublic.com.com/ - 54k - Sep 29, 2005 - Cached - Similar pages

TechRepublic - **Real** World, **Real Time**, **Real** IT
Incident, Problem and **Event Management**: The IT Process Wars! Live on Wednesday, Sept. ... TechRepublic - **Real** World. **Real Time**. **Real** IT. ...
techrepublic.com.com/?tag=ft - 55k - Cached - Similar pages

Real-Time Conferencing: Guide to Internet Conferencing, Web ...
Real-Time Conferencing Software and Services: Comprehensive, independent guide to ... streaming media, and a variety of online **event management** solutions. ...
www.thinkofit.com/webconf/**realtime**.htm - 73k - Cached - Similar pages

SYSVIEW **Realtime** Performance **Management Event** Capture Option ...
Unicenter CA-SYSVIEW **Realtime** Performance **Management Event** Capture Option. Monitoring and Managing Mainframe System Resource Performance ...
www3.ca.com/Solutions/ProductOption.asp?ID=3393 - 24k - Cached - Similar pages

SYSVIEW **Realtime** Performance **Management**, Operations **Management**
Unicenter CA-SYSVIEW **Realtime** Performance **Management Event** Capture Option Monitoring and Managing Mainframe System Resource Performance ...
www3.ca.com/Solutions/Product.asp?ID=1524 - 26k - Cached - Similar pages

Scheduling software & calendar software for room scheduling ...
Schedule and **manage** your conference rooms, A/V equipment, personnel, catering, ... you can provide **event** updates in **real time** to your audience via your ...
www.meetingmaker.com/ - 20k - Cached - Similar pages

Sherlock Enterprise Security **Management** Platform - **Real-time** ...
... clients to co-monitor enterprise-wide security **events** in **real-time**, ... It integrates **management**, monitoring, scanning and threat intelligence data and ...
www.lurhq.com/soc.html - 38k - Cached - Similar pages

DBTA: In-Depth - March 2005
The answer is **real time** data **management** and it's not only critical to program trading; ... **Real time** systems require new, **event** processing technology. ...
www.dbta.com/in-depth/mar05/rugg-palmer.html - 35k - Cached - Similar pages

CGI - Solutions - **Real-time** Operations **Management** Solution (ROMS)
Vital to the modern utilities operation, ROMS **real-time** integration between ... and **Event** Manager - Centralized Information Hub for Network **Event Management** ...
www.cgi.com/web/en/library/ solutions/utilities/74049.htm - 39k - Sep 29, 2005 - Cached - Similar pages

Dynamic CPU **Management** for **Real-Time**, Middleware-Based Systems

Dynamic CPU **Management** for **Real-Time**, Middleware-Based Systems ... The CPU Broker mediates between multiple **real-time** tasks and the facilities of a ...
www.cs.utah.edu/flux/papers/cpubroker-rtas04-base.html - 6k - Cached - Similar pages

Gooooooooooogle

Result Page: **Previous** 1 2 3 4 5 6 7 8 9 10 11 **Next**

event management and real time Search

Search within results | Language Tools | Search Tips

Google Home - Advertising Programs - Business Solutions - Hurricane Resources - About Google

Web Images Groups News Froogle Local more »

event management and real time capturing vid Search Advanced Search Preferences

The "AND" operator is unnecessary -- we include all search terms by default. [details]

Web Results **1** - **10** of about **6,060,000** for **event management and real time capturing video image**. (**0.50** se

Tip: Looking for pictures? Try Google Images

Videoguys Vegas **Video** - Sony Vegas 6+DVD Production Suite $499.95 ...
Apply **image**-correction fi lters to fi ne-tune still **images** and **video**. ...
Play back DVD projects in **real-time** on a virtual DVD player while simultaneously ...
www.**video**guys.com/vegas**video**.html - 50k - Sep 29, 2005 - Cached - Similar pages

Videoguys MPEG2 / DVD Page
The technology is now here to put your home / corporate / **event videos** onto DVD!
... Only Vegas® 6 software combines **real-time** SD, DV, and HDV **video** editing ...
www.**video**guys.com/mpeg2.htm - 100k - Sep 29, 2005 - Cached - Similar pages

Microsoft Announces Breakthrough High-Definition **Video** Production ...
... filmmakers, and corporate and **event** videographers on the Windows XP desktop."
HD-SDI **Capture** and **Real-Time**, Multistream, High-Definition **Video** Editing ...
go.microsoft.com/fwlink/?LinkId=27935 - 26k - Cached - Similar pages

MacInTouch: timely news and tips about the Apple Macintosh
This release adds a Dock menu (Mac OS X only), screenshot **events**, ... will have
a mix of everything from small text files to large **video capture** files. ...
www.macintouch.com/index.shtml - Similar pages

[PDF] The Streaming Data **Management** Challenge
File Format: PDF/Adobe Acrobat - View as HTML
site available in **real time**. over the shipboard. network. **Event** Logging ...
station monitors the **video** stream and initiates **image**. **capture** upon request from ...
www.ropos.com/documents/datamgmt.pdf - Similar pages

[PDF] IRL: An Interactive **Real-Time** Logging System for ROVs
File Format: PDF/Adobe Acrobat - View as HTML
site available in **real time**. over the shipboard. network. **Event** logging ...
of this station monitors the **video** stream and initiates **image capture** upon ...
www.ropos.com/documents/IRLpaper.pdf - Similar pages

NetBotz - Remote Environmental Monitoring, Early Detection ...
The **events** and **captured video** data area stored within the clip database on NetBotz
Central ... Ability to view multiple cameras in **real-time** in Live View ...
www.netbotz.com/cmpns/surveillanceFAQ.html - 31k - Cached - Similar pages

Windows Marketplace: Webcams & **Video**
Also it is possible to use a media file instead of **video capture** device. ...
offers support and tools for **real-time** streaming **video** and motion detection, ...
www.windowsmarketplace.com/ results.aspx?bcatid=762&page=4 - 43k - Cached - Similar pages

GV-1240 series **video capture** boards
Quality gv-1240 **video capture** boards and computer based **video** security systems
... The DSP display board displays **real-time** (30fps) **video** on all cameras and ...
www.cctvsentry.com/GV-900_series.htm - 39k - Cached - Similar pages

Olympus MIC-D: Digital **Image Capture** and Processing
This software panel can be employed to **capture** single **images**, **time**-lapse sequence, or full-motion **video** sequences and save these **image** or audiovisual files ...
www.olympusmicro.com/micd/**image**processing/ - 34k - Cached - Similar pages

Goooooooooogle ▶
Result Page: 1 2 3 4 5 6 7 8 9 10 **Next**

event management and real time ca | Search

Search within results | Language Tools | Search Tips | Dissatisfied? Help us improve

Google Home - Advertising Programs - Business Solutions - Hurricane Resources - About Google

Capturing Real-Time Requiremen

By Bruce Powel Douglass
Embedded Systems Programming
(11/01/01, 09:37:50 AM EST)

Requirements are too often co-mingled with design element way to focus on capturing only the essentials, with UML.

Many developers regard requirements capture with a disdain norm for Windows crashes and Richard Simmons exercise videos. They s of time that diverts them from what they ought to be doing: crank However, in a requirements-driven process, the developers always what they're doing actually relates to the goals and purposes of th

To properly understand what features ought to be designed and im well as how they ought to work, it is necessary to have a deep und the following concepts: the purposes of the system; the workflow (applicable) with respect to the system; the set of features the syst the devices with which it must interact and how those interactions what should happen when something expected or "bad" occurs; an the features must be visible to the user and the external devices. T part of the requirements or specification of the system. If you unde requirements thoroughly, your development work will be more pro have less reworking to do, and your customers will be happy.

In a requirements-centric development, all work relates in some w requirements specification of the system. Early in analysis, we try how the system fits into its environment (including the user). Soor detailing exactly which features we want the system to provide to work in that environment and exactly how we want those features elements in the system's environment. Later, we design the intern system to meet those specifications, and finally we construct test v system to ensure the appropriate level of completeness, fidelity, ar

I find that real-time and embedded developers often have difficult requirements from design. The chosen design is usually just one o of meeting the requirements. Many bright and experienced develo the design aspect so ingrained in them that they find this distinctio have developed an approach for understanding, capturing, and ma requirements based on my work with complex projects at NASA an which is the focus of this article. This approach is part of the ROPE

Types of requirements

Just as there are two kinds of people (those who divide people into those who don't), there are two kinds of requirements: functional service. Functional requirements encompass what the system shou it should behave in a variety of circumstances. For example:

- The system shall adjust the angle of the telescope under use
- The system shall deliver anesthetic agents in gaseous form a concentration.
- Locking clamps shall engage when the elevator cable breaks.
- The device shall alarm if the heart rate falls below 30 beats

Quality of service (QoS) requirements specify how well a functiona shall be accomplished. In real-time and embedded systems, QoS r may specify properties of the system (for example, range, speed, t capacity, reliability, maintainability, evolvability, time to market, s predictability, schedulability), or properties of the process. As a rul it's something that can be quantified or optimized, then it is a QoS For example (QoS requirements italicized):

- The angle of the telescope shall be set in units of 0.1 degree maximum error of 0.01 degrees.
- The anesthetic agent shall be controllable from 0.00% to 5.0 in units of 0.01% with an accuracy of 0.005%.
- Locking clamps shall engage in the event of an elevator supp

breakage within less than 0.5 seconds.

- The device shall alarm within 10 seconds if the heart rate fall beats per minute.

The defining characteristic of real-time systems is the level to whic requirements figure into the correctness of the system. In non-rea late is acceptable. In real-time systems, late is unacceptable. Put a real-time system is not necessarily fast, but it is predictably timely real-time systems may be hard real-time, which means that respo for aperiodic systems or actions taken when a periodic task begins systems) must complete by a specified deadline.

Systems may also be soft real-time. For example:

- Event responses shall be handled on average within a certair
- A certain number of event responses shall be handled within timeframe.
- A specified failure rate is permitted.

Because the mathematics required to analyze soft real-time systen difficult than for the simpler, hard real-time case, it is very commo real-time systems as hard real-time to simplify the analysis.[2] Th approach is an overdesign of the system, with, typically, an increas recurring cost due to the overdesigned hardware platform.

In my approach, functional requirements are modeled as use cases specifications, actions, and message sequences. QoS requirements as constraints of some kind, applied against one or more functiona

Use cases

A use case is a named coherent collection of related requirements around system capability. A use case is large-scale, typically corres three to 10 pages of textual requirements. Use cases define little ir specific requirements per se, but they serve as a way to organize a them. A good use case:

- Focuses on the user's or actor's perspective of the system (n implementation of its interfaces or its internals)
- Captures a closely related set of requirements
- Returns a result visible to one or more actors
- Does not reveal or imply system internal structure or implem
- Is independent from other use cases and may be concurrent
- Consists of a set of messages exchanged between the systen more actors (more than just one!)

Relationships among use cases can be used, but there's a caveat: newcomers to use case modeling use these relationships to do a fu decomposition of the system's internal structure; this is not what t The purpose of use case relations is to depict relations among thes requirements. The most common relations are specializations (ster specific) of the dependency relation (shown using a dashed line wi

arrowhead). The <<includes>> relation means that a larger use ca smaller one. For example, a use case for a spacecraft might be "Ta a planet" and another might be "Send information to Earth-side St Executing each of these use cases involves rolling the spacecraft tc orientation-either to point the camera at the planet or to aim the a Earth. Thus, they could both <<includes>> a smaller use case, su Attitude."

<<extends>> is similar to <<includes>> except that the smaller optional and only used in certain situations. For example, suppose commands sent to a spacecraft could potentially lead to a loss of tl You might want user validation and authorization guaranteed befor such commands. In this case, the larger "Process Ground Comman might be extended by a "Validate User."

Additionally, one use case may be more general or specific than an example, there may be multiple ways to do a Validate User use cas Authorization Code, Validate by Fingerprint Scan, or Validate by Vc Recognition. Each of these is a specialized form of the general Vali case.

We will use these relations in a very specific way when we capture for large complex systems.

Detailed requirements

Since a use case is a container of detailed requirements, just provi of the use case isn't enough. We need to provide the details. In th process we call this "detailing the use case."

There are two primary means to detail a use case-by example or b By far, the most common is by example. This is done by constructi scenarios of message exchange between the system and the actor associated with that use case. This approach has advantages and c The advantages include the simplicity of the representation and th which non-technical stakeholders can understand how the system respect to the use case. The disadvantages include the fact that a represented by an infinite set of scenarios; the number that is actu must be trimmed down somehow. Also, there is typically no way tc when you give an example. That is, there is no way to specify proh behaviors.

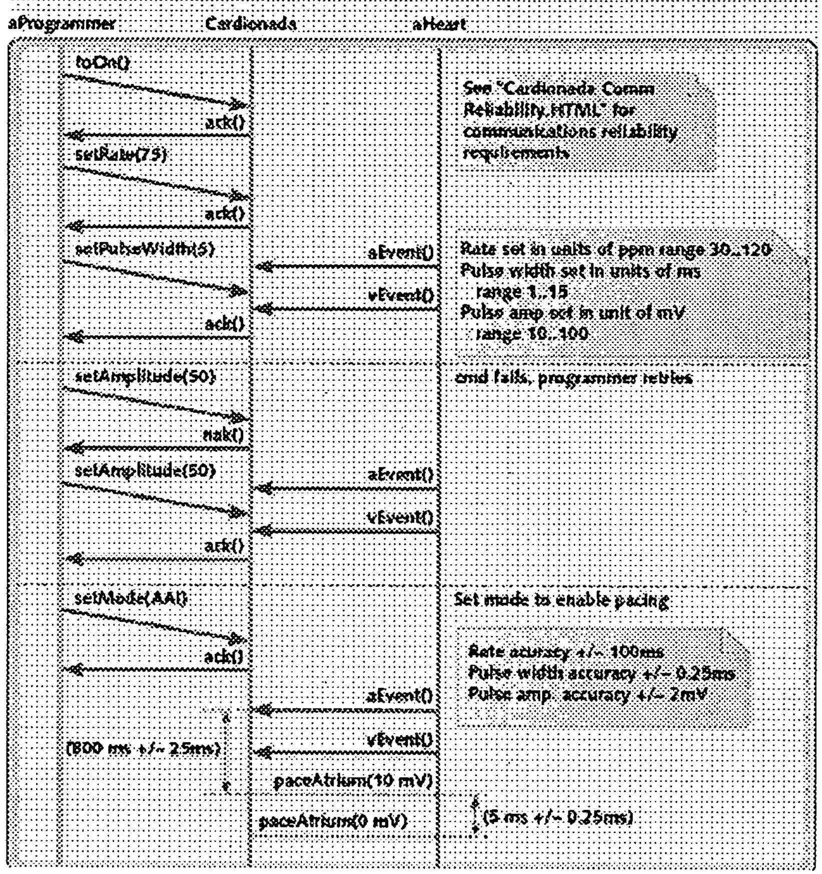
Detailing a use case by specification gets around these disadvanta a single location for the details that applies to each of the infinite s It can also state prohibitions as requirements. On the downside, pa formal languages (such as statecharts) are used to specify requirei digit IQ is required, which may disallow certain managers and mar understanding the requirements. My recommendation is to general we will see later.

Scenarios and message sequence charts

A scenario is a specific path through a use case. The most commor of a scenario is a message sequence chart, as shown in Figure 1. T

are called instance lines, and at the system specification level, the) actors and either the use case or the system fulfilling the role of th prefer to use the use case because it helps me identify the context particular scenario. Note that at this level, we do not include objec system. Looking ahead, later we will add internal objects to our sc(how our designs actually meet our requirements, but they should r system-level use case scenarios. The goal at this point is to captur(not design.

Figure 1: Scenario example



A typical system might have anywhere from half a dozen to a doze and each use case might have half a dozen to several dozen scena Since there is an infinite set from which the scenarios can be drawı decide which ones to explicitly represent? The ROPES process guid add scenarios to a use case only when they demonstrate or depict more new requirements. You're done when you can't come up with that add a new requirement.

Functional requirements are shown on sequence diagrams as orde| sequences. That is, you're showing that a particular sequence of m be allowed. If the order within a message set is unimportant, you (constraint {unordered} to the set of messages. QoS requirements constraints that attach to the instance lines, individual messages, (messages. The most common constraints are timeliness constraint applied to an ordered pair of messages. In Figure 1, a timing const down at the bottom using a common notation: a vertical line betw(horizontal bars marking points in time on the scenario. Other QoS

shown in note boxes on the right of the diagram.

Specifications for requirements capture

The other primary approach to detailing requirements is to do it by Either informal or formal languages can be used, or a combination informal languages, we usually mean written specifications. Some elaborate fields used to specify the use case. For example, Schneic suggest:[3]

- Use case name
- Actors
- Priority (project)
- Status (project)
- Preconditions
- Postconditions
- Extension points
- Included use cases
- Flow of events
- List of related diagrams (sequence, statechart, activity, and :

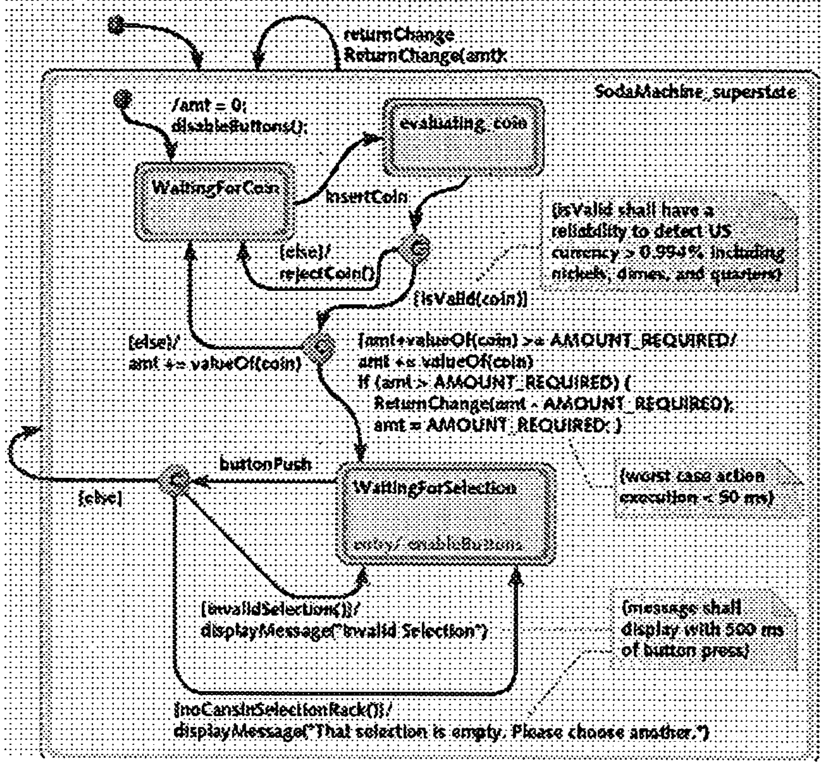
Of these, I feel only the preconditions and postconditions are requi things are shown using other views (such as the diagrams themsel

For formal languages, the UML provides the statechart and its cous chart. Statecharts are most applicable when the use case has state distinguishable conditions of existence as defined by a set of event accepted, behaviors performed, and reachability of subsequent sta use case is in State A, it accepts a certain set of messages and eve certain set of behaviors, and can reach a finite set of other states. distinguishable from other such states in that one or more of these different. When an autopilot is executing "Controlling Flight Path," certain things it can and cannot do when taking off vs. when in cru states.

Activity charts are just a specialized form of a statechart. Activity c when the primary means to transition from one state to the next d completion of the actions executed within a state rather than upon explicit message or event from somewhere else.

Consider a soda pop machine with two actors (the Customer and tl Rep). Let's focus on a Deliver Soda Can use case. It is difficult to ir individually all the possible ways in which users might insert coins buttons to get a can of soda from the machine, even without the a the price. However, it is relatively straightforward to do so using a shown in Figure 2.

Figure 2: Soda machine use case statechart



The statechart in the figure has only four states to manage the trai user inserting coins and selecting the desired flavor of soda.[4] All directly relevant to the specification of the use case are shown on t (although not their implementation). Notice also that no internal ol identified, but some data are: specifically, amt tracks how much th entered, and AMOUNT_REQUIRED is the cost of a single can of sod various operations used within the actions, but it isn't at all impliec there are or how they relate to each other.

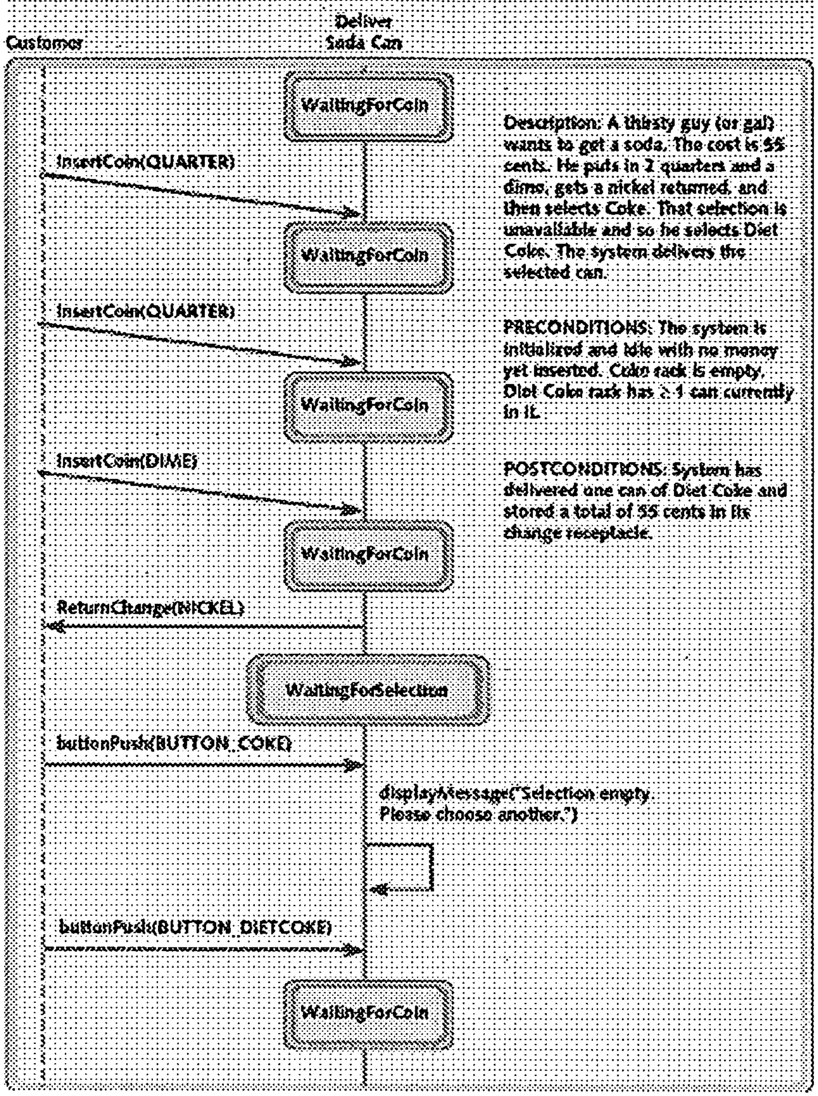
In fact, a set of objects will realize this use case (that's UML-speak "implement"). All that we can be sure of is that, in any correct desi specified will collectively be able to provide the services as specifie statechart in Figure 2.

In the final analysis, either statecharts or activity diagrams can be specification of requirements.

Relating specifications and scenarios

When you use a formal language, such as statecharts, to specify a are capturing the entire infinite set of scenarios all in one place. A nothing more than a particular path through the statechart. For ex shows one particular scenario represented by this statechart. In th cost of the soda is 55 cents. The customer puts in two quarters an receives a nickel in change. Then he selects Coke, but there is no r the machine displays a message to that effect. The customer then Coke and the system delivers it. Notice that some of the relevant s state machine are shown on the use case instance line-this aids th relating the scenario back to the statechart specification.

Figure 3: Thirsty guy scenario



Of course, there are other paths through the statechart; these are
scenarios. In general, you will want to construct the set of scenaric
statechart. You do this by making a different scenario for every dif
through the statechart, although you'll only want to do the looping
and representative examples of the concurrent regions (and-states

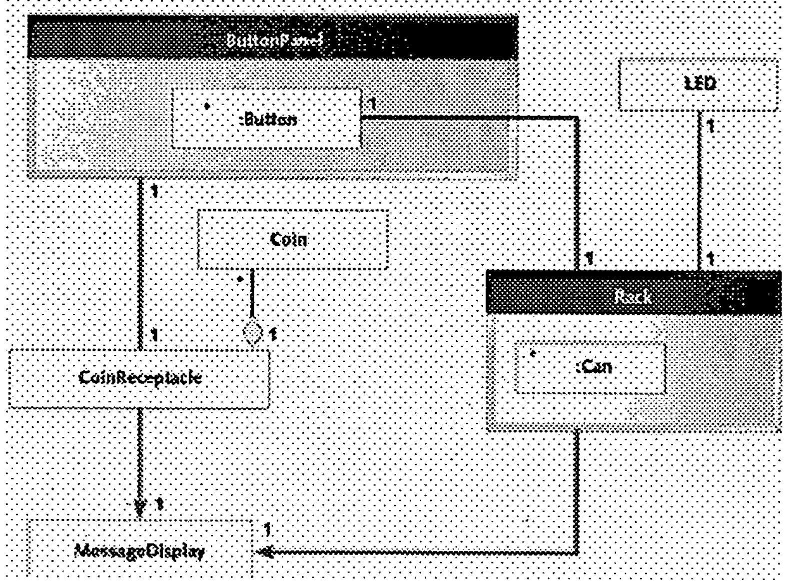
Moving from requirements into design

As mentioned earlier, a use case is ultimately realized by a set of c
together to provide the necessary behavior. This set of objects is c
collaboration in UML. It has a specific notation (a dashed oval) that
commonly used. Most often, the use case collaboration is shown as
diagram, showing the relevant classes of the objects that participa
in the collaboration.

Getting a good set of objects can be tricky, as it is not at all obviou
case model. In the ROPES process, you use object identification str
identify the object participating in the collaboration. The ROPES pr
about a dozen different strategies which, while different and distin

the objects they find to a significant degree. Commonly, you will a|
four different strategies simultaneously to identify all objects in the
Using such an approach, one could come up with an object model s
shown in Figure 4.

Figure 4: Soda machine collaboration class diagram

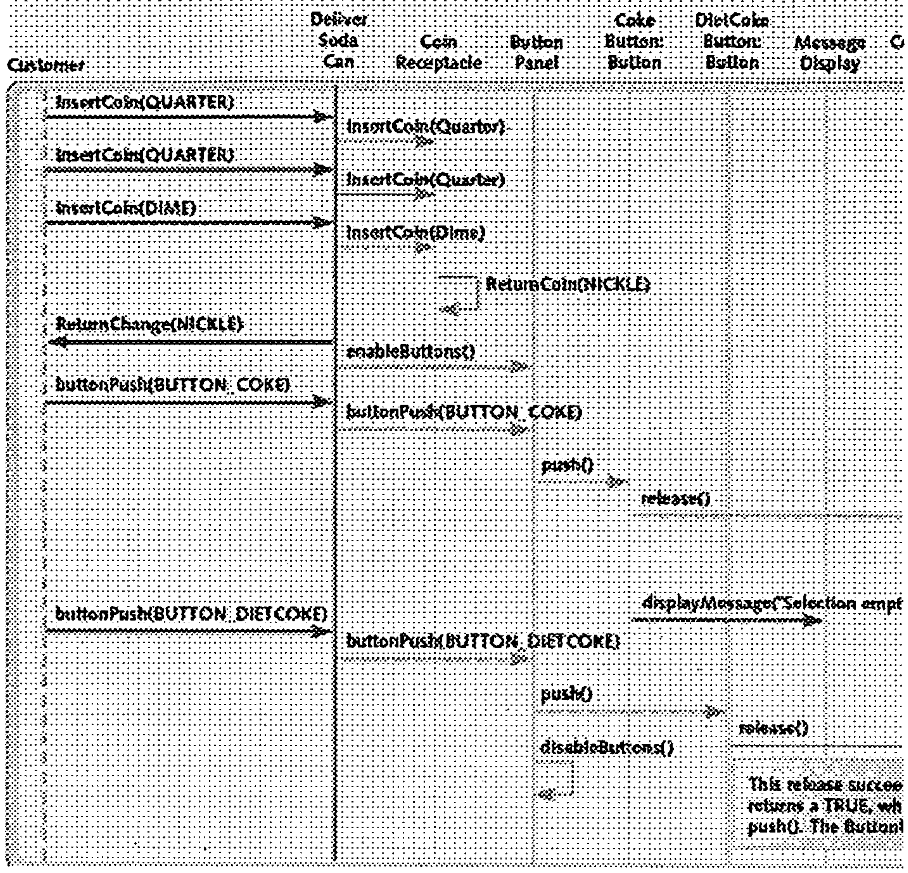


The really important question is how do you ensure and demonstra design model really does realize the use case model? The answer i: execution. We evaluate and demonstrate the adequacy of a design executing that design model. Specifically, we want to execute the \ scenarios we used to state requirements using the newly added de If the design can execute all of the requirements then we've done , can't, then we need to fix our design model.

To do this, first we come up with a set of scenarios at the system ι where the players are the actors and The System (or the system p of a specific use case). To test the adequacy of a design, we do the scenarios but now we add the design elements to the scenario and exchange of messages among them necessary to fully realize the t

Figure 5 is the same scenario shown in Figure 3, but we've added t collaborative elements from the class diagram.[5] We can walk thı scenario and see how the objects inside the system collaborate to ı

Figure 5: Thirsty guy scenario with design detail



This approach of validation via execution can be applied at any levı abstraction. As we add more detail to a highly complex system witl components, and composite classes, we can be sure that we're doi when, at the specified level of abstraction, it executes all of the scε for the use case.

This approach means that even if you use statecharts to specify re you will also draw scenarios. Since a statechart can have loops anc regions, it can represent an infinite set of scenarios. Which scenari draw and trace through the levels of design abstraction? Again, the process has an answer: draw a scenario for every non-looping patl or-states and each looping path exactly once; for concurrent regioı representative interleaving of the concurrent regions.

Caught

We've discussed an approach to the capturing of requirements for embedded systems that has been shown to be effective in projects small. Use cases, as defined in the UML, are used as an organizing managing requirements and a combination of scenarios and specifi capture the detailed requirements. For large systems, a system en captures the subsystem architecture and maps the level-0 use cası allows teams to go off and work independently with some assuranc subsystems will properly fit into the system when they're all done.

Bruce Powel Douglass has over 20 years of experience designin

real-time applications in a variety of hard real-time environments. authored several books, including *Real-Time UML* and *Doing Hard* *Developing Real-Time Systems with UML, Objects, Frameworks an* Bruce worked with methodologists at Rational and other companie UML specification. Bruce is Chief Evangelist at I-Logix, and can be bpd@ilogix.com.

Endnotes

1. Douglass, Bruce Powel. "On the ROPES," Embedded Systems Pr December 2000, p. 140.
Back

2. There is the old saying that hard real-time systems are hard, bu systems are harder, meaning that the accurate characterization of systems is technically much more difficult.
Back

3. Schneider, Geri and Jason P. Winters. Applying Use Cases: A Pr New York: Addison-Wesley, 1998.
Back

4. Douglass, Bruce Powel. "UML Statecharts," Embedded Systems January 1999, p. 22.
Back

5. Remember that scenarios show instances, not types (that is, cla scenario has two different Rack instances, for example, but the cla shows only a single Rack class.
Back

Return to November 2001 Table of Contents

Capturing Real-Time Requiremen

By Bruce Powel Douglass
Embedded Systems Programming
(11/01/01, 09:37:50 AM EST)

Requirements are too often co-mingled with design element way to focus on capturing only the essentials, with UML.

Many developers regard requirements capture with a disdain norm for Windows crashes and Richard Simmons exercise videos. They s of time that diverts them from what they ought to be doing: crank However, in a requirements-driven process, the developers always what they're doing actually relates to the goals and purposes of th

To properly understand what features ought to be designed and im well as how they ought to work, it is necessary to have a deep und the following concepts: the purposes of the system; the workflow c applicable) with respect to the system; the set of features the syst the devices with which it must interact and how those interactions what should happen when something expected or "bad" occurs; an the features must be visible to the user and the external devices. T part of the requirements or specification of the system. If you unde requirements thoroughly, your development work will be more pro have less reworking to do, and your customers will be happy.

In a requirements-centric development, all work relates in some w requirements specification of the system. Early in analysis, we try how the system fits into its environment (including the user). Soor detailing exactly which features we want the system to provide to work in that environment and exactly how we want those features elements in the system's environment. Later, we design the intern system to meet those specifications, and finally we construct test v system to ensure the appropriate level of completeness, fidelity, ar

I find that real-time and embedded developers often have difficulty requirements from design. The chosen design is usually just one of of meeting the requirements. Many bright and experienced develop the design aspect so ingrained in them that they find this distinctio have developed an approach for understanding, capturing, and ma requirements based on my work with complex projects at NASA an which is the focus of this article. This approach is part of the ROPE!

Types of requirements

Just as there are two kinds of people (those who divide people into those who don't), there are two kinds of requirements: functional a service. Functional requirements encompass what the system shou it should behave in a variety of circumstances. For example:

- The system shall adjust the angle of the telescope under use
- The system shall deliver anesthetic agents in gaseous form a concentration.
- Locking clamps shall engage when the elevator cable breaks.
- The device shall alarm if the heart rate falls below 30 beats p

Quality of service (QoS) requirements specify how well a functiona shall be accomplished. In real-time and embedded systems, QoS r may specify properties of the system (for example, range, speed, t capacity, reliability, maintainability, evolvability, time to market, sa predictability, schedulability), or properties of the process. As a rul it's something that can be quantified or optimized, then it is a QoS For example (QoS requirements italicized):

- The angle of the telescope shall be set in units of 0.1 degree: maximum error of 0.01 degrees.
- The anesthetic agent shall be controllable from 0.00% to 5.0 in units of 0.01% with an accuracy of 0.005%.
- Locking clamps shall engage in the event of an elevator supp

breakage within less than 0.5 seconds.

- The device shall alarm within 10 seconds if the heart rate fall beats per minute.

The defining characteristic of real-time systems is the level to whic requirements figure into the correctness of the system. In non-rea late is acceptable. In real-time systems, late is unacceptable. Put a real-time system is not necessarily fast, but it is predictably timely real-time systems may be hard real-time, which means that respo for aperiodic systems or actions taken when a periodic task begins systems) must complete by a specified deadline.

Systems may also be soft real-time. For example:

- Event responses shall be handled on average within a certair
- A certain number of event responses shall be handled within timeframe.
- A specified failure rate is permitted.

Because the mathematics required to analyze soft real-time systen difficult than for the simpler, hard real-time case, it is very commo real-time systems as hard real-time to simplify the analysis.[2] Th approach is an overdesign of the system, with, typically, an increa: recurring cost due to the overdesigned hardware platform.

In my approach, functional requirements are modeled as use case: specifications, actions, and message sequences. QoS requirements as constraints of some kind, applied against one or more functiona

Use cases

A use case is a named coherent collection of related requirements around system capability. A use case is large-scale, typically corre: three to 10 pages of textual requirements. Use cases define little ir specific requirements per se, but they serve as a way to organize a them. A good use case:

- Focuses on the user's or actor's perspective of the system (n implementation of its interfaces or its internals)
- Captures a closely related set of requirements
- Returns a result visible to one or more actors
- Does not reveal or imply system internal structure or implem
- Is independent from other use cases and may be concurrent
- Consists of a set of messages exchanged between the systen more actors (more than just one!)

Relationships among use cases can be used, but there's a caveat: newcomers to use case modeling use these relationships to do a fu decomposition of the system's internal structure; this is not what t The purpose of use case relations is to depict relations among thes requirements. The most common relations are specializations (ster specific) of the dependency relation (shown using a dashed line wi

arrowhead). The <<includes>> relation means that a larger use ca smaller one. For example, a use case for a spacecraft might be "Ta a planet" and another might be "Send information to Earth-side St Executing each of these use cases involves rolling the spacecraft tc orientation-either to point the camera at the planet or to aim the a Earth. Thus, they could both <<includes>> a smaller use case, su Attitude."

<<extends>> is similar to <<includes>> except that the smaller optional and only used in certain situations. For example, suppose commands sent to a spacecraft could potentially lead to a loss of tl You might want user validation and authorization guaranteed befor such commands. In this case, the larger "Process Ground Comman might be extended by a "Validate User."

Additionally, one use case may be more general or specific than an example, there may be multiple ways to do a Validate User use cas Authorization Code, Validate by Fingerprint Scan, or Validate by Vc Recognition. Each of these is a specialized form of the general Vali case.

We will use these relations in a very specific way when we capture for large complex systems.

Detailed requirements

Since a use case is a container of detailed requirements, just provi of the use case isn't enough. We need to provide the details. In the process we call this "detailing the use case."

There are two primary means to detail a use case-by example or b By far, the most common is by example. This is done by constructi scenarios of message exchange between the system and the actor associated with that use case. This approach has advantages and c The advantages include the simplicity of the representation and the which non-technical stakeholders can understand how the system respect to the use case. The disadvantages include the fact that a represented by an infinite set of scenarios; the number that is actu must be trimmed down somehow. Also, there is typically no way tc when you give an example. That is, there is no way to specify proh behaviors.

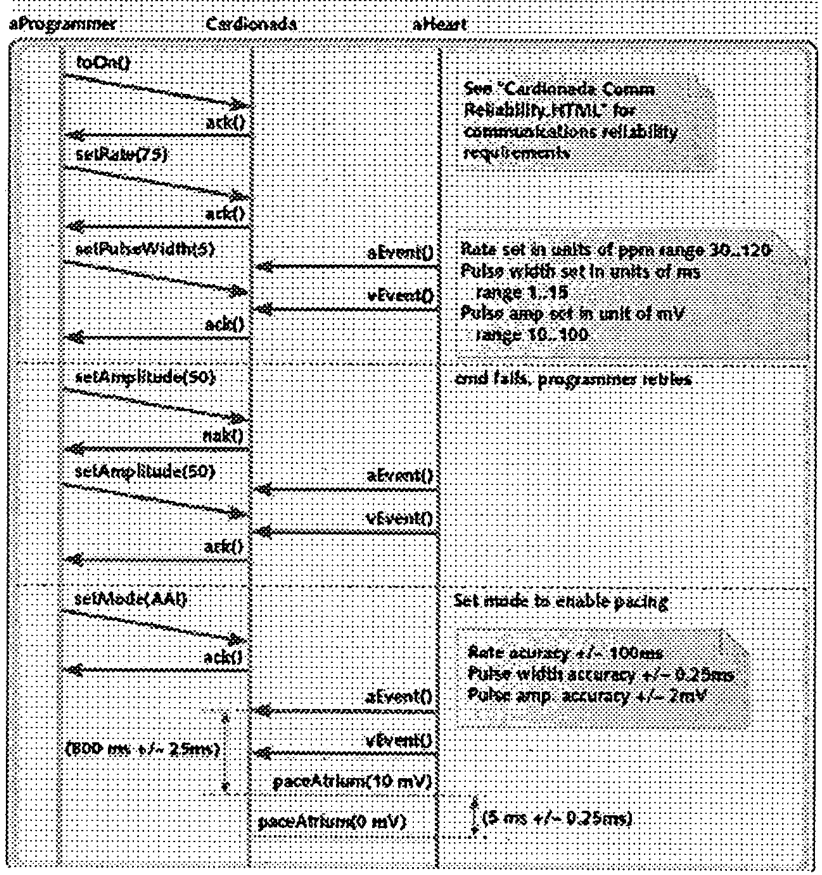
Detailing a use case by specification gets around these disadvanta a single location for the details that applies to each of the infinite s It can also state prohibitions as requirements. On the downside, pa formal languages (such as statecharts) are used to specify requirer digit IQ is required, which may disallow certain managers and mar understanding the requirements. My recommendation is to general we will see later.

Scenarios and message sequence charts

A scenario is a specific path through a use case. The most commor of a scenario is a message sequence chart, as shown in Figure 1. T

are called instance lines, and at the system specification level, they
actors and either the use case or the system fulfilling the role of th
prefer to use the use case because it helps me identify the context
particular scenario. Note that at this level, we do not include objec
system. Looking ahead, later we will add internal objects to our sc
how our designs actually meet our requirements, but they should r
system-level use case scenarios. The goal at this point is to captur
not design.

Figure 1: Scenario example



A typical system might have anywhere from half a dozen to a doze
and each use case might have half a dozen to several dozen scena
Since there is an infinite set from which the scenarios can be drawi
decide which ones to explicitly represent? The ROPES process guid
add scenarios to a use case only when they demonstrate or depict
more new requirements. You're done when you can't come up with
that add a new requirement.

Functional requirements are shown on sequence diagrams as order
sequences. That is, you're showing that a particular sequence of m
be allowed. If the order within a message set is unimportant, you c
constraint {unordered} to the set of messages. QoS requirements
constraints that attach to the instance lines, individual messages, c
messages. The most common constraints are timeliness constraint
applied to an ordered pair of messages. In Figure 1, a timing const
down at the bottom using a common notation: a vertical line betwe
horizontal bars marking points in time on the scenario. Other QoS

shown in note boxes on the right of the diagram.

Specifications for requirements capture

The other primary approach to detailing requirements is to do it by Either informal or formal languages can be used, or a combination informal languages, we usually mean written specifications. Some elaborate fields used to specify the use case. For example, Schneic suggest:[3]

- Use case name
- Actors
- Priority (project)
- Status (project)
- Preconditions
- Postconditions
- Extension points
- Included use cases
- Flow of events
- List of related diagrams (sequence, statechart, activity, and

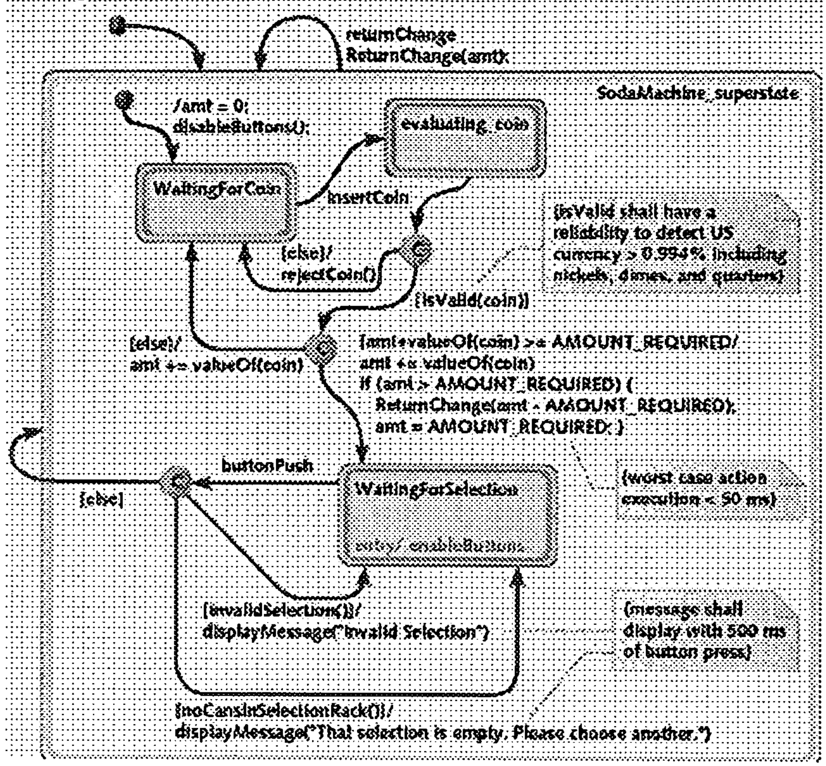
Of these, I feel only the preconditions and postconditions are requi things are shown using other views (such as the diagrams themsel

For formal languages, the UML provides the statechart and its cous chart. Statecharts are most applicable when the use case has state distinguishable conditions of existence as defined by a set of event accepted, behaviors performed, and reachability of subsequent sta use case is in State A, it accepts a certain set of messages and eve certain set of behaviors, and can reach a finite set of other states. distinguishable from other such states in that one or more of these different. When an autopilot is executing "Controlling Flight Path," certain things it can and cannot do when taking off vs. when in cru states.

Activity charts are just a specialized form of a statechart. Activity c when the primary means to transition from one state to the next d completion of the actions executed within a state rather than upon explicit message or event from somewhere else.

Consider a soda pop machine with two actors (the Customer and tl Rep). Let's focus on a Deliver Soda Can use case. It is difficult to ir individually all the possible ways in which users might insert coins buttons to get a can of soda from the machine, even without the a the price. However, it is relatively straightforward to do so using a shown in Figure 2.

Figure 2: Soda machine use case statechart



The statechart in the figure has only four states to manage the trai
user inserting coins and selecting the desired flavor of soda.[4] All
directly relevant to the specification of the use case are shown on t
(although not their implementation). Notice also that no internal ol
identified, but some data are: specifically, amt tracks how much th
entered, and AMOUNT_REQUIRED is the cost of a single can of sod
various operations used within the actions, but it isn't at all impliec
there are or how they relate to each other.

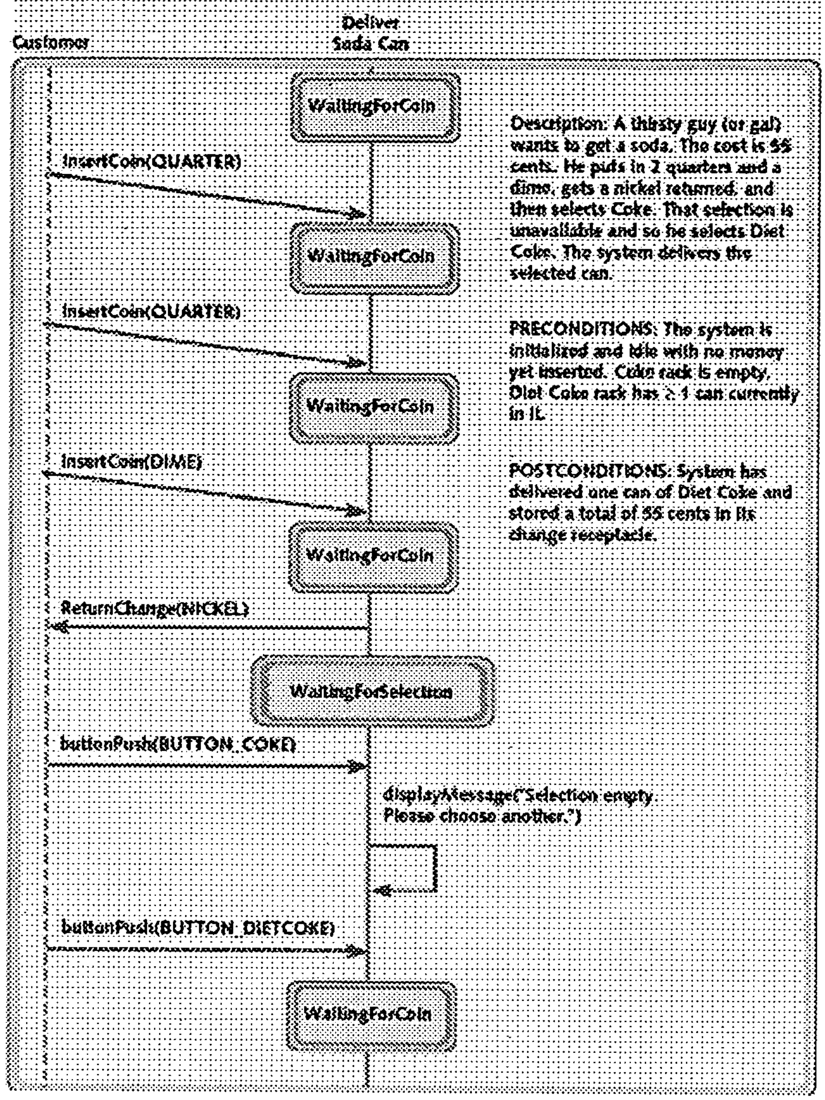
In fact, a set of objects will realize this use case (that's UML-speak
"implement"). All that we can be sure of is that, in any correct desi
specified will collectively be able to provide the services as specifie
statechart in Figure 2.

In the final analysis, either statecharts or activity diagrams can be
specification of requirements.

Relating specifications and scenarios

When you use a formal language, such as statecharts, to specify a
are capturing the entire infinite set of scenarios all in one place. A
nothing more than a particular path through the statechart. For ex
shows one particular scenario represented by this statechart. In th
cost of the soda is 55 cents. The customer puts in two quarters an
receives a nickel in change. Then he selects Coke, but there is no r
the machine displays a message to that effect. The customer then
Coke and the system delivers it. Notice that some of the relevant s
state machine are shown on the use case instance line-this aids th
relating the scenario back to the statechart specification.

Figure 3: Thirsty guy scenario



Of course, there are other paths through the statechart; these are scenarios. In general, you will want to construct the set of scenaric statechart. You do this by making a different scenario for every dif through the statechart, although you'll only want to do the looping and representative examples of the concurrent regions (and-states

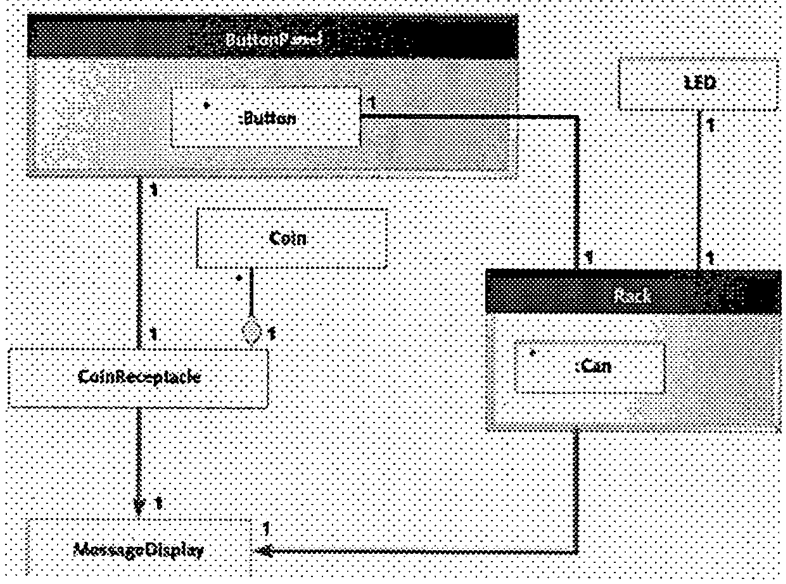
Moving from requirements into design

As mentioned earlier, a use case is ultimately realized by a set of c together to provide the necessary behavior. This set of objects is c collaboration in UML. It has a specific notation (a dashed oval) thai commonly used. Most often, the use case collaboration is shown as diagram, showing the relevant classes of the objects that participa in the collaboration.

Getting a good set of objects can be tricky, as it is not at all obviou case model. In the ROPES process, you use object identification str identify the object participating in the collaboration. The ROPES pr about a dozen different strategies which, while different and distin

the objects they find to a significant degree. Commonly, you will a| four different strategies simultaneously to identify all objects in the Using such an approach, one could come up with an object model : shown in Figure 4.

Figure 4: Soda machine collaboration class diagram

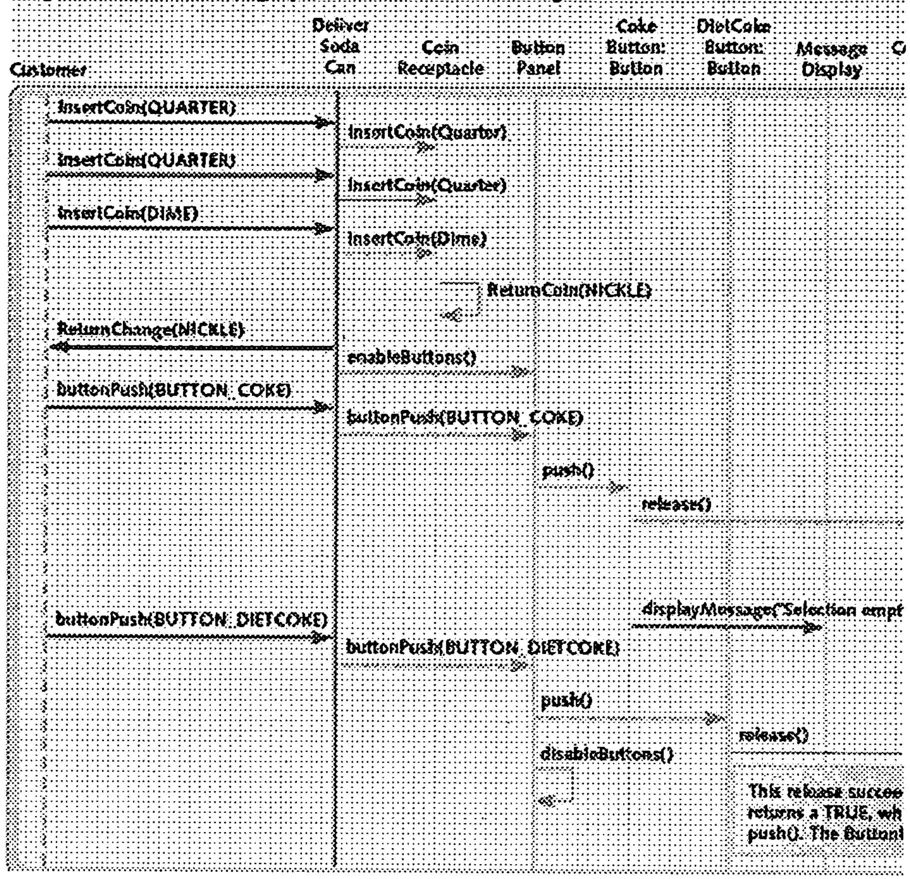


The really important question is how do you ensure and demonstra design model really does realize the use case model? The answer i: execution. We evaluate and demonstrate the adequacy of a design executing that design model. Specifically, we want to execute the scenarios we used to state requirements using the newly added de If the design can execute all of the requirements then we've done can't, then we need to fix our design model.

To do this, first we come up with a set of scenarios at the system where the players are the actors and The System (or the system p of a specific use case). To test the adequacy of a design, we do the scenarios but now we add the design elements to the scenario and exchange of messages among them necessary to fully realize the b

Figure 5 is the same scenario shown in Figure 3, but we've added t collaborative elements from the class diagram.[5] We can walk thi scenario and see how the objects inside the system collaborate to

Figure 5: Thirsty guy scenario with design detail



This approach of validation via execution can be applied at any lev abstraction. As we add more detail to a highly complex system witl components, and composite classes, we can be sure that we're doi when, at the specified level of abstraction, it executes all of the sce for the use case.

This approach means that even if you use statecharts to specify re you will also draw scenarios. Since a statechart can have loops anc regions, it can represent an infinite set of scenarios. Which scenari draw and trace through the levels of design abstraction? Again, the process has an answer: draw a scenario for every non-looping patl or-states and each looping path exactly once; for concurrent regio representative interleaving of the concurrent regions.

Caught

We've discussed an approach to the capturing of requirements for embedded systems that has been shown to be effective in projects small. Use cases, as defined in the UML, are used as an organizing managing requirements and a combination of scenarios and specifi capture the detailed requirements. For large systems, a system en captures the subsystem architecture and maps the level-0 use cas allows teams to go off and work independently with some assuranc subsystems will properly fit into the system when they're all done.

Bruce Powel Douglass has over 20 years of experience designin

real-time applications in a variety of hard real-time environments. authored several books, including *Real-Time UML* and *Doing Hard* *Developing Real-Time Systems with UML, Objects, Frameworks an* Bruce worked with methodologists at Rational and other companie UML specification. Bruce is Chief Evangelist at I-Logix, and can be bpd@ilogix.com.

Endnotes

1. Douglass, Bruce Powel. "On the ROPES," Embedded Systems Pr December 2000, p. 140.
Back

2. There is the old saying that hard real-time systems are hard, bu systems are harder, meaning that the accurate characterization of systems is technically much more difficult.
Back

3. Schneider, Geri and Jason P. Winters. Applying Use Cases: A Pra New York: Addison-Wesley, 1998.
Back

4. Douglass, Bruce Powel. "UML Statecharts," Embedded Systems January 1999, p. 22.
Back

5. Remember that scenarios show instances, not types (that is, cla scenario has two different Rack instances, for example, but the cla shows only a single Rack class.
Back

Return to November 2001 Table of Contents

Google

Web Images Groups News Froogle Local more »

event management and real time capturing vid Search Advanced Search Preferences

The "AND" operator is unnecessary -- we include all search terms by default. [details]

Web Results **21** - **30** of about **6,060,000** for **event management and real time capturing video image**. **(0.11 s**

Tip: Looking for pictures? Try Google Images

Virtual Reality Platform
Video Inputs: AVI, MPEG; **Images** Inputs: PSD... **Real-time** Simulation of title navigation ... texture **management**, graphical user interface tools, input **event** ...
www.inrialpes.fr/sed/PRV/CATALOG/software.html - 34k - Cached - Similar pages

Convera Sees Increased Federal Demand for Digital Content ...
Space Shuttle and International Space Station **video** content is **captured**, encoded, archived and managed to provide **real-time** and post mission analysis and ...
www.convera.com/press/pr_032001b.asp - 19k - Cached - Similar pages

Embedded.com - **Capturing Real-Time** Requirements
Here's a way to focus on **capturing** only the essentials, with UML. ... Systems may also be soft **real-time**. For example:. **Event** responses shall be handled on ...
www.embedded.com/story/OEG20011016S0126 - 94k - Cached - Similar pages

PacketHop > News **Events** > Press Releases > 2005 > 071105
Respondents also highly regarded **real-time** multicast **video**, global positioning system ... Whiteboarding enables users to "mark up" a **captured video image**, ...
www.packethop.com/news_events/ press_releases/2005/071105.php - 14k - Cached - Similar pages

[PDF] Code:DSP **Video**, **Image** & Signal Processing Seminar
File Format: PDF/Adobe Acrobat - View as HTML
use of FPGAs in **video**, **image** and signal processing applications. ... Generate FPGA design with **real-time video capture** and filtering ...
www.altera.com/education/**events**/ northamerica/code_dsp05/dsp-na-agenda.pdf - Similar pages

[PDF] Experience the Latest in Network **Video** Technology
File Format: PDF/Adobe Acrobat - View as HTML
event management, built-in **video** motion. detection and a complete security ... Superior **image** quality and **real-time**. **video images** at up to 30 frames per ...
www.axis.com/files/newsletters/at_axis_mar04_intl.pdf - Similar pages

CommsDesign - **Real-time video capture** creates memory power concerns
Video capture and 3-D gaming at VGA resolution and 6-megapixel **image capture** will all become ... This is particularly the case for **real-time video capture**. ...
www.commsdesign.com/design_corner/ showArticle.jhtml?articleID=22101850 - 44k - Cached - Similar pages

Hydra: A robust and self managing **video** sensing system for ...
These systems collate the string of **video events** that were part of the particular ... The **captured** scenes can either be analyzed in **real-time** or stored for ...
www.cse.nd.edu/~csesys/hydra/ - 27k - Cached - Similar pages

Instant Messaging Graduates To Full **Real-Time** Collaboration ...
Real-time Video Chat - **Video** Filter Effects - **Image Capture** - **Video Capture** ...
X-Events · Products & Services · Knowledge **Management** · Products & Services ...
www.masternewmedia.org/news/2004/ 11/17/instant_messaging_graduates_to_full.htm - 61k - Sep 30, 2005 -

Cached - Similar pages

GV-1480 series **video capture** board
Quality gv-1480 **video capture** boards and computer based **video** security systems ... The DSP display board displays **real-time** (30fps) **video** on all cameras and ...
www.cctvsentry.com/GV-1000_series.htm - 38k - Cached - Similar pages

Gooooooooooogle
Result Page: **Previous** 1 2 3 4 5 6 7 8 9 10 11 12 **Next**

event management and real time ca Search

Search within results | Language Tools | Search Tips

Google Home - Advertising Programs - Business Solutions - Hurricane Resources - About Google

01 O

-> Design Corner

Real-time video capture creates memory power concerns

By Ivan Greenberg
EE Times
Jun 28, 2004

PRINT TH STORY

As carriers and content providers usher in the wireless lifestyle, handsets wil unprecedented transformation. Video capture and 3-D gaming at VGA resolu megapixel image capture will all become commonplace in the next two years

Until recently, mobile processors, LCD displays, image sensors and RF power amplifiers have been the focus of system designers. However, with the advent of new applications like 3-D gaming, designers are becoming increasingly aware of the energy consumed by the handsets memory subsystem.

Tomorrow's handsets will shuffle voluminous amounts of media between processor and memory subsystem, creating a new power hot spot in memory. Unlike yesterday's phone, whose internal data transfer was fundamentally limited to protocol stack processing, next-generation handsets will perform advanced signal processing on two-dimensional data s 4, H.264, JPEG2000 and 3-D image processing. Additionally, business applic as Excel, PowerPoint, Word and Outlook will become commonplace on tomor phones.

To meet these requirements, memory suppliers are looking at the expected and use model patterns of tomorrow's media-rich handset in order to formul

techniques for reducing energy consumption of this memory subsystem. As a handset system designers have a plethora of low-power memory platforms f to choose. These platforms are typically offered in multichip packages to con handset real estate and combine NOR, SRAM, pseudoSRAM and Nand produc

Most of today's high-end phones use NOR flash and DRAM. Many smart phor feature phones feature two NOR devices-one for code storage and code exec the other for data storage. Additionally, a single mobile DRAM device is used scratch pad for temporary storage of images and execution of media process algorithms.

Given the state of mobile multimedia, NOR flash has served the handset plat However, screen resolution, image resolution and video resolution are trendi phenomenal rates, mandating nonvolatile memories with ultrafast storage ba and ultralow power consumption. Given the benefits of Nand technology, thi: render the choice of NOR a non sequitur. This is particularly the case for real capture.

Video capture and record

The data flow requirements of video are quite different than those for image Video compression requires constant DRAM read/write access, as the video c captured occurs over several minutes, or in some cases, a single hour. DRAM power must be accounted for because it plays a large role in overall energy consumption.

While the benefits of mobile DRAM over PC DRAM are obvious in this applicat of mobile double data-rate RAM (DDR) over mobile single data-rate RAM (SD elusive. By reducing the clock frequency of the mobile DDR device by a facto device's active current can be reduced to half that of a mobile SDR device de same bandwidth. This results in a 35 percent reduction in power consumptio compared to a mobile SDR device.

For the calculations below, assume a user takes 10 15-minute video clips du course of a week. Further, assume that the user captured these clips at VGA using a high-quality MPEG4 encoder with an average bit rate of 1.3 Mbits/se DRAM active current, we can assume single-bank operation.

Unlike image capture, a single 8-Mbyte bank is sufficient for holding frames, kernel and video-processing work space. Video capture and encode at VGA r requires on the order of 100 Mbytes/s sustained DRAM bandwidth.

For this article, let's assume a DRAM device that is capable of delivering 400 when clocked at 100 MHz. Assuming 50 percent efficiency for the application memory controller, systems using this DRAM should be able to sustain 100 M bandwidth with a 50 percent read/write duty cycle. Duty cycle is defined her ratio of the average active DRAM time, or read/write accesses, to average D time, or refresh mode:

Usable DDR BW = (DDR controller efficiency) x (DDR read/write duty cycle) bandwidth)

= (50 percent) x (50 percent) x (400 Mbit/s) = 100 Mbits/s

- Network Systems DesignLine
- Power Management DesignLine
- Programmable Logic DesignLine
- Video [Imaging DesignLine
- Wireless Net DesignLine

ELECTRONICS GROUP SITES

- NEW! SpecSearch
- eeProductCenter
- Electronics Supply & Manufacturing
- Inside [DSP]
- Conferences and Events
- Electronics Supply & Manufacturing--China
- Electronics Express
- NetSeminar Services

The DRAM energy consumed is calculated using the effective duty cycle abov

mA-hrDRAM_active = [(Isingle_bank_active) x (duty cycle) x (15-minute vic (number of clips/week)]/3,600

mA-hrDRAM_active = (45 mA) x (50 percent) x (15 x 60) x (10)/3,600 = 57

While NOR's program rate can handle a video bit rate of 1.3 Mbits/s, it must total of 150 minutes storing the video during our contrived test week. Using below for the case of video capture, we arrive at:

mA-hrNOR = [(number of video-clips/week) x (Iprogram) x (program time p minute video clip)]/3,600

mA-hrNOR = (10) x (36 mA) x (975 seconds)/3,600 = 98 mA-hr

If we assume same program time for NAND (150 minutes), energy consump Nand will be 22 percent that for NOR since Nand program current is 8 mA, v 36 mA. This would yield a Nand energy consumption of approximately 22 m/ However, algorithm developers can reduce this further by buffering up comp in DRAM for deferred ultra-fast storate to Nand. By doing this, program time 150 min of video (1.4 Gbytes) approaches that achieved if one were writing Mbytes/s.

Since the video bit rate is so low, energy consumed buffering up compressec DRAM is a function of self-refresh current (approximately 0.15 mA)]. Compa current consumed during Nand program operation (8 mA), it becomes clear battle against time, buffering in DRAM has a tremendous advantage.

The equations below are used to determine energy consumption for Nand de DRAM devices with the assumption that system program time approaches th using a 4-Mbyte/s Nand device.

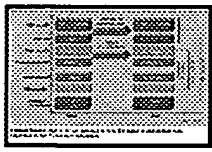
mA-hr512 Mbit_Nand = (10) x (8 mA) x (37 seconds)/3,600 = 0.82 mA-hr

mA-hr256 Mbit_DRAM = (10) x (0.15 mA) x (900)/3,600 = 0.375 mA-hr

Taking the results from all of the equation above, the power consumption fo NOR/mobile DDR subsystem comes in at 155 mA-hr, while that for a Nand/n memory subsystem measures 58 mA-hr.

While the advantages of Nand/DRAM-based systems shown above are impre new wireless frontier will stimulate the sagest memory suppliers to innovate areas. In the quest for lower power, we can expect future Nand/DRAM comb feature enhanced interfaces, novel packaging and innovative architectural el Regardless of the innovative path, one thing is certain: The mobile memory play a pivotal role in reducing power and enhancing the user's experience of handsets.

Ivan K. Greenberg (igreenberg@ssi.samsung.com)is director of strategic ma Samsung Semiconductor (San Jose, Calif.).



See related chart

Home | Register | About | Feedback | Contact